

1 What is claimed is:

2
3 1. A method of operating a plurality of virus checkers for on-demand anti-virus
4 scanning concurrent with on-access anti-virus scanning, the method comprising:

5 combining on-demand anti-virus scan requests and on-access anti-virus scan
6 requests in a virus scan request queue; and

7 distributing the on-demand anti-virus scan requests and the on-access anti-virus
8 scan requests from the virus scan request queue to the virus checkers.

9
10 2. The method as claimed in claim 1, wherein the on-access anti-virus scan requests
11 are produced in response to user access to files.

12
13 3. The method as claimed in claim 1, wherein the on-demand anti-virus scan
14 requests are produced in response to a system administrator requesting a scan of files
15 within a specified file system.


16
17 4. The method as claimed in claim 1, wherein a pool of threads distribute the on-
18 demand anti-virus scan requests and the on-access anti-virus scan requests from the virus
19 scan request queue to the virus checkers, each anti-virus scan request on the virus scan
20 request queue being serviced by a respective one of the threads in the pool of threads.

21
22 5. The method as claimed in claim 1, wherein the on-access anti-virus scan requests
23 are given priority over the on-demand anti-virus scan requests by inhibiting the

1 placement of on-demand anti-virus scan requests onto the virus scan request queue when
2 the number of anti-virus scan requests on the virus scan request queue reaches a
3 threshold, and not inhibiting the placement of on-access anti-virus scan requests onto the
4 virus scan request queue when the number of anti-virus scan requests on the virus scan
5 request queue reaches the threshold.

6
7 6. The method as claimed in claim 1, which includes grouping the on-demand anti-
8 virus scan requests into chunks of multiple ones of the on-demand anti-virus scan
9 requests, and placing the chunks onto the virus scan request queue.

10
11 7. The method as claimed in claim 5, which includes inhibiting the placement of at
12 least one of the chunks onto the virus scan request queue until completion of anti-virus
13 scanning for the anti-virus scan requests in a prior one of the chunks.

14
15 8. A method of operating a plurality of virus checkers, the method comprising: 
16 distributing on-demand anti-virus scan requests and on-access anti-virus scan
17 requests to the virus checkers so that the virus checkers perform on-demand anti-virus
18 scanning concurrent with on-access anti-virus scanning;

19 which includes grouping the on-demand anti-virus scan requests into chunks of
20 multiple ones of the on-demand anti-virus scan requests, and for each chunk, distributing
21 the multiple ones of the on-demand anti-virus scan requests over the virus checkers.

1 9. The method as claimed in claim 8, wherein the on-access anti-virus scan requests
2 are produced in response to user access to files.

3
4 10. The method as claimed in claim 8, wherein the on-demand anti-virus scan
5 requests are produced in response to a system administrator requesting a scan of files
6 within a specified file system.

7
8 11. The method as claimed in claim 8, which includes inhibiting the distribution of
9 the multiple ones of the on-demand anti-virus scan requests from at least one of the
10 chunks to the virus checkers until completion of anti-virus scanning for the anti-virus
11 scan requests in a prior one of the chunks.

12
13 12. A method of operating a plurality of virus checkers for on-demand anti-virus
14 scanning concurrent with on-access anti-virus scanning, the method comprising:

15 combining on-demand anti-virus scan requests and on-access anti-virus scan
16 requests in a virus scan request queue; and

17 a pool of threads distributing the on-demand anti-virus scan requests and the on-
18 access anti-virus scan requests from the virus scan request queue to the virus checkers,
19 each anti-virus scan request on the virus scan request queue being serviced by a
20 respective one of the threads in the pool of threads,

21 which includes grouping the on-demand anti-virus scan requests into chunks of
22 multiple ones of the on-demand anti-virus scan requests, and for each chunk, checking
23 whether the number of anti-virus scan requests on the virus checking queue is less than a

1 threshold, and upon finding that the number of anti-virus scan requests on the virus
2 checking queue is less than the threshold, placing said each chunk on the virus scan
3 request queue.

4
5 13. The method as claimed in claim 12, wherein the on-access anti-virus scan
6 requests are produced in response to user access to files.

7
8 14. The method as claimed in claim 12, wherein the on-demand anti-virus scan
9 requests are produced in response to a system administrator requesting a scan of files
10 within a specified file system.

11
12 15. The method as claimed in claim 12, which includes inhibiting the placement of at
13 least one of the chunks onto the virus scan request queue until completion of anti-virus
14 scanning for the anti-virus scan requests in a prior one of the chunks.

15
16 16. A virus checking system comprising: /
17 a plurality of virus checkers for on-demand anti-virus scanning concurrent with
18 on-access anti-virus scanning;
19 a virus scan request queue; and
20 at least one processor coupled to the virus checkers and the virus scan request
21 queue for sending virus scan requests from the virus scan request queue to the virus
22 checkers, said at least one processor being programmed for placing on-demand anti-virus
23 scan requests and on-access anti-virus scan requests onto the virus scan request queue,

1 and for distributing the on-demand anti-virus scan requests and the on-access virus scan
2 requests from the virus scan request queue to the virus checkers.

3
4 17. The virus checking system as claimed in claim 16, wherein said at least one
5 processor and said virus scan request queue are in a file server, and the virus checkers are
6 separate from the file server.

7
8 18. The virus checking system as claimed in claim 16, wherein said at least one
9 processor is programmed to place each on-access request onto the virus scan request
10 queue in response to user access of a respective file.

11
12 19. The virus checking system as claimed in claim 16, wherein said at least one
13 processor is programmed to produce the on-demand anti-virus scan requests in response
14 to a system administrator requesting a scan of files within a specified file system.

15
16 20. The virus checking system as claimed in claim 16, wherein said at least one
17 processor is programmed to execute multiple threads for distributing the on-demand anti-
18 virus scan requests and the on-access anti-virus scan requests from the virus scan request
19 queue to the virus checkers, each anti-virus scan request on the virus scan request queue
20 being serviced by a respective one of the threads in the pool of threads.

21
22 21. The virus checking system as claimed in claim 16, wherein said at least one
23 processor is programmed for giving the on-access anti-virus scan requests priority over

1 the on-demand anti-virus scan requests by inhibiting the placement of on-demand anti-
2 virus scan requests onto the virus scan request queue when the number of anti-virus scan
3 requests on the virus scan request queue reaches a threshold, and not inhibiting the
4 placement of on-access anti-virus scan requests onto the virus scan request queue when
5 the number of anti-virus scan requests on the virus scan request queue reaches the
6 threshold.

7
8 22. The virus checking system as claimed in claim 16, wherein said at least one of the
9 processors is programmed for grouping the on-demand anti-virus scan requests onto
10 chunks of multiple ones of the on-demand anti-virus scan requests, and placing the
11 chunks onto the virus scan request queue.

12
13 23. The virus checking system as claimed in claim 22, which includes inhibiting the
14 placement of at least one of the chunks onto the virus scan request queue until completion
15 of anti-virus scanning for the anti-virus scan requests in a prior one of the chunks.

16
17 24. A virus checking system comprising: /
18 a plurality of virus checkers for on-demand anti-virus scanning concurrent with
19 on-access anti-virus scanning; and
20 a file server coupled to the virus checkers for sending virus scan requests to the
21 virus checkers, the file server including a virus scan request queue, and the file server
22 being programmed for placing on-demand anti-virus scan requests and on-access anti-
23 virus scan requests onto the virus scan request queue; and for executing multiple threads

1 for distributing the on-demand anti-virus scan requests and the on-access anti-virus scan
2 requests from the virus scan request queue to the virus checkers, each anti-virus scan
3 request on the virus scan request queue being serviced by a respective one of the threads
4 in the pool of threads, the file server further being programmed for grouping the on-
5 demand anti-virus scan requests into chunks of multiple ones of the on-demand anti-virus
6 scan requests, and for consecutively placing the chunks onto the virus scan request queue.
7

8 25. The virus checking system as claimed in claim 24, wherein the file server is
9 programmed for producing the on-access anti-virus scan requests in response to user
10 access to files.
11

12 26. The virus checking system as claimed in claim 24, wherein the file server is
13 programmed to produce the on-demand anti-virus scan requests in response to a system
14 administrator requesting a scan of files within a specified file system.
15

16 27. The virus checking system as claimed in claim 24, wherein the file server is
17 programmed for checking for each chunk whether the number of anti-virus scan requests
18 on the virus checking queue is less than a threshold, and upon finding that the number of
19 anti-virus scan requests on the virus checking queue is less than the threshold, placing
20 said each chunk on the virus scan request queue.
21

22 28. The virus checking system as claimed in claim 24, wherein the file server is
23 programmed for inhibiting the placement of at least one of the chunks onto the virus scan

1 request queue until completion of anti-virus scanning for the anti-virus scan requests in a
2 prior one of the chunks.

3

4